IEICE TRANS. INF. & SYST., VOL.E101–D, NO.12 DECEMBER 2018

PAPER

Parametric Models for Mutual Kernel Matrix Completion

Rachelle RIVERO[†,††a)], ***Nonmember and*** **Tsuyoshi KATO**[†,†††,††††], ***Member***

SUMMARY Recent studies utilize multiple kernel learning to deal with incomplete-data problem. In this study, we introduce new methods that do not only complete multiple incomplete kernel matrices simultaneously, but also allow control of the flexibility of the model by parameterizing the model matrix. By imposing restrictions on the model covariance, overfitting of the data is avoided. A limitation of kernel matrix estimations done via optimization of an objective function is that the positive definiteness of the result is not guaranteed. In view of this limitation, our proposed methods employ the LogDet divergence, which ensures the positive definiteness of the resulting inferred kernel matrix. We empirically show that our proposed restricted covariance models, employed with LogDet divergence, yield significant improvements in the generalization performance of previous completion methods.

key words: *multiview learning, incomplete-data problem, probabilistic PCA, factor analysis, kernel matrix completion*

1. Introduction

Since the seminal work of Lanckriet et al. [1], data fusion has become an integral part of data analysis especially in the field of computational biology and bioinformatics. For instance, they showed that a set of proteins can be described by a number of relevant data sources, such as protein-protein interaction, gene expression, and amino-acid sequences. This is because relevant data sources provide complementary perspectives or "views" of the objects and, together, these pieces of information present a bigger picture of the relations the objects have with each other. This notion of exploiting the multiple views of the data for better learning is more commonly known as *multi-view learning*. The data sources, however, may come in various forms (e. g. strings, trees, or graphs), and kernel methods [2], [3] provide a way of integrating such heterogeneous data by transforming them into a common format: as kernel matrices. A Bayesian formulation for efficient multiple kernel learning was presented by Gönen [4], while early works in computational biology utilized multi-view learning to classify protein functions [1], [5]–[7].

A shortcoming of multi-view learning, however, is incomplete data. Incomplete data is relatively common in almost all researches, no matter how well-designed the experiments or the data gathering methods are. A few examples of incomplete data occurrences are: a sensor may suddenly fail and go off in a remote sensing experiment; participants may not have answered some questions in a questionnaire; and inevitable data acquisition error, among others. Analysis of incomplete data may lead to invalid conclusions, since they only give minimal insights about the objects at hand.

Thus, in addition to dealing with the heterogeneity of the data, kernel methods are utilized in several studies to handle missing information. A data source with some missing information leads to an incomplete kernel matrix (i. e., a matrix with missing entries); however, complete kernel matrices derived from complete data sources can be exploited to provide solutions to the incomplete-data problem. Studies addressing this problem via kernel methods have progressed over time—from completion of a kernel matrix through a single complete kernel matrix [8], and through multiple complete kernel matrices [9]—to simultaneous completion of multiple incomplete kernel matrices [10], [11]. The kernel completion technique in [10] associates the kernel matrices to the covariance of a zero-mean Gaussian distribution, and employs the expectation-maximization (EM) algorithm [12] to minimize the objective function. On the other hand, the technique in [11] learns reconstruction weights to express a particular incomplete kernel matrix as a convex combination of the other kernel matrices. Although these two methods tackle a similar setting, the main difference between them is that [11] employ Euclidean metric to assess the distance between kernel matrices, which requires additional constraints to keep all kernel matrices positive definite. In [10], LogDet divergence [13], [14] is employed, and this not only keeps the positive definiteness automatically but also brings a strong connection to the classical approach of estimating missing values in vectorial data. With the missing entries inferred, the completed kernel matrices can now be fused and utilized for tasks such as multi-view clustering and classification.

In the previous solution to the task of multiple kernel matrix completion [10], a model matrix is introduced as a representative kernel matrix of the given multiple kernel matrices. The model matrix is allowed to move to any point in

Manuscript received April 18, 2018.
Manuscript revised July 27, 2018.
Manuscript publicized September 26, 2018.

[†]The authors are with Domain of Electronics and Informatics, Mathematics and Physics, Graduate School of Science and Technology, Gunma University, Kiryu-shi, 376–8515 Japan.
[††]The author is with Institute of Mathematics, College of Science, University of the Philippines, Diliman 1101, Quezon City, Philippines.
[†††]The author is with Center for Research on Adoption of NextGen Transportation Systems (CRANTS), Gunma University, Kiryu-shi, 376–8515 Japan.
[††††]The author is with Integrated Institute for Regulatory Science, Waseda University, Tokyo, 162–0041 Japan.
a) E-mail: rrivero@math.upd.edu.ph
DOI: 10.1587/transinf.2018EDP7139

the positive definite cone which is a very broad manifold in the set of symmetric matrices. Like the classical model fitting task, a too flexible model tends to overfit to the given empirical data. The flexibility of the model should be adjusted to make the model generalize well, but is impossible to do in the previous model.

In view of this, we present alternative approaches to the previous method for multiple kernel matrix completion by defining parametric models that can move only on a submanifold in the positive definite cone. Both the previous and the new methods can be related to a statistical framework. The previous method can be explained with maximum likelihood estimation of a full covariance Gaussian, which often tends to overfit the data due to large degrees of freedom. On the other hand, the proposed methods can be associated to a parametric model that imposes a restriction to the covariance matrix parameter. The number of degrees of freedom in the new models can be adjusted, thereby improving the generalization performance.

2. Problem Setting

Suppose that we have ℓ objects, $x_1, \ldots, x_\ell$, and K data sources. For example, if we want to analyze a set of proteins, then the number of proteins is ℓ, and the data sources may be amino-acid sequences, gene expressions, protein cubic structures, and so on. Let $\kappa_k(\cdot,\cdot)$ be a kernel function between two proteins for k-th data source. For amino-acid sequence, the corresponding kernel function $\kappa_k(\cdot,\cdot)$ can be a positive definite sequence similarity. For gene expressions, the corresponding kernel function $\kappa_k(\cdot,\cdot)$ can be the RBF kernel between gene expression data. Let $\boldsymbol{Q}^{(k)}$ be the $\ell \times \ell$ kernel matrix for k-th data source where its (i, j)-th entry is given by $Q^{(k)}_{i,j} = \kappa_k(x_i, x_j)$.

The problem setting discussed in this paper is the particular setting where some or all of the data sources have missing information. For example, the cubic structures of some proteins have not been determined. Gene expressions are observed for only a subset of the ℓ genes on the microarray chip. Given these, the rows and columns in the kernel matrix corresponding to the objects with missing data have missing entries (Fig. 1). Now, suppose that for the k-th data source, only the data for the n_k objects are available. We denote by $\boldsymbol{Q}^{(k)}_{vh,vh}$ the k-th kernel matrix $\boldsymbol{Q}^{(k)}$ in which the order of rows and columns are rearranged so that the first n_k available objects are followed by the remaining $(\ell - n_k)$ objects with unavailable data. This rearrangement of the rows and the columns results in the following symmetric partitioned matrix:

$$\boldsymbol{Q}^{(k)}_{vh,vh} = \begin{pmatrix} \boldsymbol{Q}^{(k)}_{v,v} & \boldsymbol{Q}^{(k)}_{v,h} \\ \boldsymbol{Q}^{(k)}_{h,v} & \boldsymbol{Q}^{(k)}_{h,h} \end{pmatrix}, \tag{1}$$

where $\boldsymbol{Q}^{(k)}_{v,v} \in \mathbb{S}^{n_k}_{++}$, with $\mathbb{S}^{n_k}_{++}$ denoting the set of $n_k \times n_k$ strictly positive definite symmetric matrices. The algorithm then mutually infers the (missing) entries for the submatrices $\boldsymbol{Q}^{(k)}_{v,h} \in \mathbb{R}^{n_k \times (\ell - n_k)}$, $\boldsymbol{Q}^{(k)}_{h,v} = \left(\boldsymbol{Q}^{(k)}_{v,h}\right)^\top$, and $\boldsymbol{Q}^{(k)}_{h,h} \in \mathbb{S}^{(\ell-n_k)}_{++}$, for $k = 1, \ldots, K$.

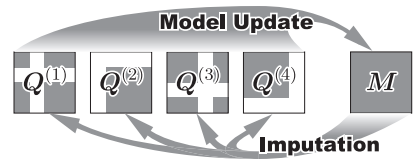

Fig. 1 Overview of mutual kernel matrix completion methods. In this figure, four incomplete empirical kernel matrices, $\boldsymbol{Q}^{(1)}, \ldots, \boldsymbol{Q}^{(4)}$, are assumed to be given. Here, the shaded areas in the empirical kernel matrices pertain to the objects with available data, whereas the white areas pertain to the objects with unavailable data. A model matrix $\boldsymbol{M}$ is introduced, and the proposed method repeats two steps: model update step and imputation step. In model update step, the model matrix $\boldsymbol{M}$ is fitted to the set of the current empirical kernel matrices $(\boldsymbol{Q}^{(k)})_{k=1}^4$. In imputation step, the missing entries in each of the empirical kernel matrices are estimated using the current $\boldsymbol{M}$.

3. FC-MKMC: Existing Model

In the previous study [10], an algorithm for mutual kernel matrix completion had already been developed. Henceforth, we will refer to the previous method as the full covariance mutual kernel matrix completion (FC-MKMC), and review the method in this section. To infer the missing values in the incomplete kernel matrices, FC-MKMC introduces an $\ell \times \ell$ model matrix $\boldsymbol{M}$, and finds the set of kernel matrices $\boldsymbol{Q}^{(k)}$ that are as close to each other as possible through the model matrix $\boldsymbol{M}$. The objective function of FC-MKMC is the sum of LogDet divergences [13], [14]:

$$J_{\mathrm{FC}}(\mathcal{H}, \boldsymbol{M}) := \sum_{k=1}^{K} \mathrm{LogDet}\left(\boldsymbol{Q}^{(k)}, \boldsymbol{M}\right), \tag{2}$$

where $\mathcal{H} := \left\{\boldsymbol{Q}^{(k)}_{v,h}, \boldsymbol{Q}^{(k)}_{h,h}\right\}_{k=1}^{K}$ is the set of submatrices containing the missing entries, and $\boldsymbol{M}$ is the model matrix. The LogDet divergence is defined as

$$\begin{aligned} \mathrm{LogDet}\left(\boldsymbol{Q}^{(k)}, \boldsymbol{M}\right) := \frac{1}{2}\Big(&\log\det \boldsymbol{M} - \log\det \boldsymbol{Q}^{(k)} \\ &+ \left\langle \boldsymbol{M}^{-1}, \boldsymbol{Q}^{(k)} - \boldsymbol{M} \right\rangle\Big). \end{aligned} \tag{3}$$

An advantage of using LogDet divergence is that a necessary property for valid kernel matrices, the positive definiteness, is ensured for the resultant completed kernel matrices. The approach of FC-MKMC is essentially similar to the well-known probabilistic approach for classical incomplete data completion [15], where missing values in incomplete vectorial data are to be inferred. In the approach for the classical task, a probabilistic model is introduced to be fitted to the temporarily completed data, and the missing values are imputed with the most probable values using the current inference of the probabilistic model. The number of degrees of freedom of a probabilistic model provides an important perspective for the success or for the failure of data completion: too rigid models cannot capture the underlying data distribution, while too flexible models are often overfitted to the data set. In FC-MKMC, the model matrix can take any values without restriction, with $(\ell + 1)\ell/2$ degrees of freedom.

This model may be too flexible. In the next section, we shall present two new models in which the number of degrees of freedom can be adjusted.

4. Parametric Models

The model matrix in FC-MKMC is too flexible and is not tunable. Hence, we introduce two types of the model matrix: the PCA model and the FA model.

4.1 PCA-MKMC

In the PCA model, the form of the model matrix is restricted to

$$\boldsymbol{M} := \boldsymbol{W}\boldsymbol{W}^\top + \sigma^2\boldsymbol{I}, \tag{4}$$

where the matrix $\boldsymbol{W} \in \mathbb{R}^{\ell\times q}$ and scalar $\sigma^2 \in \mathbb{R}$ are the adaptive parameters of the PCA model. The number of columns in $\boldsymbol{W}$, which is q, is arbitrary. Larger q yields a more flexible model, and vice-versa. In this model, the number of degrees of freedom is $\ell q + 1 - (q-1)q/2$. The difference between the PCA model and the full-covariance model is the restriction on q and the additional term $\sigma^2\boldsymbol{I}$. It can be observed that when $q = \ell$ and $\sigma = 0$, PCA-MKMC model is reduced to FC-MKMC model.

Meanwhile, the objective function of PCA model is expressed as

$$J_{\mathrm{PCA}}(\mathcal{H}, \boldsymbol{W}, \sigma^2) := \sum_{k=1}^{K} \mathrm{LogDet}(\boldsymbol{Q}^{(k)}, \boldsymbol{W}\boldsymbol{W}^\top + \sigma^2\boldsymbol{I}). \tag{5}$$

Since the objective function is not jointly convex of the three arguments, $\mathcal{H}$, $\boldsymbol{W}$, and σ^2, the optimal solution cannot be given in closed form. Hence, we adopt the following block coordinate descent method that repeats the following two steps:

1. Imputation step:

$$\mathcal{H}^{(t)} := \operatorname*{argmin}_{\mathcal{H}} J_{\mathrm{PCA}}(\mathcal{H}, \boldsymbol{W}^{(t-1)}, \sigma^2_{t-1}); \tag{6}$$

2. Model update step:

$$(\boldsymbol{W}^{(t)}, \sigma^2_t) := \operatorname*{argmin}_{(\boldsymbol{W},\sigma^2)} J_{\mathrm{PCA}}(\mathcal{H}^{(t)}, \boldsymbol{W}, \sigma^2). \tag{7}$$

Therein, the iteration number t is the superscript of $\mathcal{H}$ and $\boldsymbol{W}$, and the subscript of σ^2. By letting $\boldsymbol{M} := \boldsymbol{W}\boldsymbol{W}^\top + \sigma^2\boldsymbol{I}$, the imputation step can be performed in the same fashion as that of FC-MKMC [10]. For each data source, the rows and the columns in $\boldsymbol{M}$ are reordered as $\boldsymbol{Q}^{(k)}_{vh,vh}$, and partitioned to obtain $\boldsymbol{M}^{(k)}_{v,v}$, $\boldsymbol{M}^{(k)}_{v,h}$, and $\boldsymbol{M}^{(k)}_{h,h}$. Using these submatrices in the model matrix, and the known submatrix in the empirical matrix $\boldsymbol{Q}^{(k)}_{v,v}$, the unknown submatrices in $\boldsymbol{Q}^{(k)}_{vh,vh}$ are re-estimated as

$$\boldsymbol{Q}^{(k)}_{v,h} := \boldsymbol{Q}^{(k)}_{v,v}\left(\boldsymbol{M}^{(k)}_{v,v}\right)^{-1}\boldsymbol{M}^{(k)}_{v,h} \tag{8}$$

and

$$\boldsymbol{Q}^{(k)}_{h,h} := \boldsymbol{M}^{(k)}_{h,h} - \boldsymbol{M}^{(k)}_{h,v}\left(\boldsymbol{M}^{(k)}_{v,v}\right)^{-1}\boldsymbol{M}^{(k)}_{v,h} + \boldsymbol{M}^{(k)}_{h,v}\left(\boldsymbol{M}^{(k)}_{v,v}\right)^{-1}\boldsymbol{Q}^{(k)}_{v,v}\left(\boldsymbol{M}^{(k)}_{v,v}\right)^{-1}\boldsymbol{M}^{(k)}_{v,h}. \tag{9}$$

Finally, the submatrices are reordered back to $\boldsymbol{Q}^{(k)}$ to obtain a new solution that minimizes $J_{\mathrm{PCA}}(\mathcal{H}, \boldsymbol{W}, \sigma^2)$ over missing values $\mathcal{H}$, with the model parameters $\boldsymbol{W}$ and σ^2 held fixed. The new value of $\boldsymbol{Q}^{(k)}$ at the t-th iteration is then denoted by $\boldsymbol{Q}^{(t,k)}$.

In the model update step, the K empirical kernel matrices are fixed, and the two model parameters, $\boldsymbol{W}$ and σ^2, are optimized. We here denote by 'const' the terms independent of $(\boldsymbol{W}, \sigma^2)$, and so the objective function can be rewritten as

$$J_{\mathrm{PCA}}(\mathcal{H}^{(t)}, \boldsymbol{W}, \sigma^2) := \frac{K}{2}\,\mathrm{logdet}(\boldsymbol{W}\boldsymbol{W}^\top + \sigma^2\boldsymbol{I}) + \frac{K}{2}\left\langle (\boldsymbol{W}\boldsymbol{W}^\top + \sigma^2\boldsymbol{I})^{-1}, \boldsymbol{S}^{(t)}\right\rangle + \mathrm{const} \tag{10}$$

where we have defined

$$\boldsymbol{S}^{(t)} := \frac{1}{K}\sum_{k=1}^{K}\boldsymbol{Q}^{(t,k)}. \tag{11}$$

Even though the missing values $\mathcal{H}$ are fixed to $\mathcal{H}^{(t)}$, the function $(\boldsymbol{W}, \sigma^2) \mapsto J_{\mathrm{PCA}}(\mathcal{H}^{(t)}, \boldsymbol{W}, \sigma^2)$ is still not convex on the space of the model parameters $(\boldsymbol{W}, \sigma^2)$. Nevertheless, surprisingly enough, the optimal solutions of the model parameters $\boldsymbol{W}$ and σ^2 are given in closed forms [16]. Let $\lambda_1, \ldots, \lambda_\ell$ be the eigenvalues of $\boldsymbol{S}^{(t)}$. Assume that $\lambda_1 \geq \cdots \geq \lambda_\ell$, and denote by $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_\ell$ their corresponding eigenvectors. It can be shown that the optimal σ^2, denoted by σ^2_t, is expressed as

$$\sigma^2_t = \frac{1}{\ell - q}\sum_{j=q+1}^{\ell}\lambda_j. \tag{12}$$

Now, let $\boldsymbol{U}_q := \left[\boldsymbol{u}_1, \ldots, \boldsymbol{u}_q\right]$ and $\boldsymbol{\Lambda}_q := \mathrm{diag}(\lambda_1, \ldots, \lambda_q)$. Here, for vector $\boldsymbol{x} \in \mathbb{R}^n$, we denote the diagonal matrix with diagonal entries $\boldsymbol{x}$ by $\mathrm{diag}(\boldsymbol{x})$. Meanwhile, $\mathrm{diag}(\boldsymbol{X})$ denotes an n-dimensional vector containing the diagonal entries in a square matrix $\boldsymbol{X} \in \mathbb{R}^{n\times n}$. And so, the optimal value of $\boldsymbol{W}$, denoted by $\boldsymbol{W}^{(t)}$, is given by

$$\boldsymbol{W}^{(t)} = \boldsymbol{U}_q(\boldsymbol{\Lambda}_q - \sigma^2_t\boldsymbol{I})^{1/2}\boldsymbol{R}, \tag{13}$$

where $\boldsymbol{R}$ is an arbitrary orthonormal matrix (i. e., $\boldsymbol{R}^\top\boldsymbol{R} = \boldsymbol{I}$).

4.2 FA-MKMC

In this section, we introduce the FA model, which is a variant of the PCA model. The FA model uses the following parametric model as the model matrix:

$$\boldsymbol{M} = \boldsymbol{W}\boldsymbol{W}^\top + \mathrm{diag}(\boldsymbol{\psi}). \tag{14}$$

The difference of this from the PCA model is the second term. In the PCA model, the second term is $\sigma^2\boldsymbol{I}$, whereas in

Algorithm 1 PCA-MKMC Algorithm.

Input: Kernel matrices $\left(\boldsymbol{Q}^{(k)}\right)_{k=1}^{K}$.
Output: Completed kernel matrices $\left(\boldsymbol{Q}^{(k)}\right)_{k=1}^{K}$.
begin
Initialize $\left(\boldsymbol{Q}^{(k)}\right)_{k=1}^{K}$ by imputing zeros in the missing entries;
Initialize the model matrix as $\boldsymbol{M} = \sum_{k=1}^{K} \boldsymbol{Q}^{(k)}/K$;
repeat
  for all $k \in \{1, \ldots, K\}$ **do**
    Use (8) and (9) to update $\boldsymbol{Q}_{v,h}^{(k)}$ and $\boldsymbol{Q}_{h,h}^{(k)}$;
  end for
  Use (12) and (13) to update $\boldsymbol{W}$ and σ^2;
  $\boldsymbol{M} := \boldsymbol{W}\boldsymbol{W}^{\top} + \sigma^2 \boldsymbol{I}$;
until convergence
end.

Algorithm 2 FA-MKMC Algorithm.

Input: Kernel matrices $\left(\boldsymbol{Q}^{(k)}\right)_{k=1}^{K}$.
Output: Completed kernel matrices $\left(\boldsymbol{Q}^{(k)}\right)_{k=1}^{K}$.
begin
Initialize $\left(\boldsymbol{Q}^{(k)}\right)_{k=1}^{K}$ by imputing zeros in the missing entries;
Initialize the model matrix as $\boldsymbol{M} = \sum_{k=1}^{K} \boldsymbol{Q}^{(k)}/K$;
repeat
  for all $k \in \{1, \ldots, K\}$ **do**
    Use (8) and (9) to update $\boldsymbol{Q}_{v,h}^{(k)}$ and $\boldsymbol{Q}_{h,h}^{(k)}$;
  end for
  Use (18) to update $\boldsymbol{W}$ and $\boldsymbol{\psi}$;
  $\boldsymbol{M} := \boldsymbol{W}\boldsymbol{W}^{\top} + \operatorname{diag}(\boldsymbol{\psi})$;
until convergence
end.

the FA model, the term can take any diagonal matrix. The number of degrees of freedom of the FA model is $\ell q + \ell - (q-1)q/2$, and the objective function is expressed as

$$\begin{aligned} &J_{\mathrm{FA}}(\mathcal{H}, \boldsymbol{W}, \boldsymbol{\psi}) \\ &:= \sum_{k=1}^{K} \operatorname{LogDet}\left(\boldsymbol{Q}^{(k)}, \boldsymbol{W}\boldsymbol{W}^{\top} + \operatorname{diag}(\boldsymbol{\psi})\right). \end{aligned} \tag{15}$$

Similar to the fitting algorithm of the PCA model, we adopt the block coordinate descent method to fit the FA model to the empirical kernel matrices. The imputation step is the same as in the PCA model. In the PCA model, when fixing $\mathcal{H}$, the optimal parameters $(\boldsymbol{W}, \sigma^2)$ can be expressed in closed form. However, in FA model, the optimal parameters $(\boldsymbol{W}, \boldsymbol{\psi})$ cannot be given in closed forms, even when $\mathcal{H}$ is fixed. In the FA model, we just improve $(\boldsymbol{W}, \boldsymbol{\psi})$ in the model update step.

The two steps in t-th iteration are summarized as follows:

1. *Imputation step*. Use (8) and (9) to infer the missing entries $\mathcal{H}^{(t)}$ such that

$$\mathcal{H}^{(t)} := \operatorname*{argmin}_{\mathcal{H}} J_{\mathrm{FA}}(\mathcal{H}, \boldsymbol{W}^{(t-1)}, \boldsymbol{\psi}^{(t-1)}); \tag{16}$$

2. *Model update step*. Update the model parameters $(\boldsymbol{W}^{(t)}, \boldsymbol{\psi}^{(t)})$ such that

$$\begin{aligned} &J_{\mathrm{FA}}(\mathcal{H}^{(t)}, \boldsymbol{W}^{(t)}, \boldsymbol{\psi}^{(t)}) \\ &\quad \leq J_{\mathrm{FA}}(\mathcal{H}^{(t)}, \boldsymbol{W}^{(t-1)}, \boldsymbol{\psi}^{(t-1)}). \end{aligned} \tag{17}$$

A new value $(\boldsymbol{W}^{(t)}, \boldsymbol{\psi}^{(t)})$ that satisfies (17) can be found as follows: Update the factor loading matrix $\boldsymbol{W}^{(t)}$ and the noise variance vector $\boldsymbol{\psi}^{(t)}$, by

$$\begin{aligned} &\boldsymbol{W}^{(t)} := \boldsymbol{S}_{xz}^{(t)} \left(\boldsymbol{S}_{zz}^{(t)}\right)^{-1}, \qquad \text{and} \\ &\boldsymbol{\psi}^{(t)} := \operatorname{diag}\left(\boldsymbol{S}^{(t)} - \boldsymbol{S}_{xz}^{(t)} (\boldsymbol{S}_{zz}^{(t)})^{-1} (\boldsymbol{S}_{xz}^{(t)})^{\top}\right), \end{aligned} \tag{18}$$

respectively, where

$$\begin{aligned} &\boldsymbol{F}^{(t)} := \left(\boldsymbol{W}^{(t-1)}\right)^{\top} \operatorname{diag}\left(\boldsymbol{\psi}^{(t-1)}\right)^{-1}, \\ &\boldsymbol{C}^{(t)} := \boldsymbol{I} + \boldsymbol{F}^{(t)} \boldsymbol{W}^{(t-1)}, \\ &(\boldsymbol{M}^{(t)})^{-1} := \operatorname{diag}(\boldsymbol{\psi}^{(t-1)})^{-1} - \left(\boldsymbol{F}^{(t)}\right)^{\top} \left(\boldsymbol{C}^{(t)}\right)^{-1} \boldsymbol{F}^{(t)}, \\ &\boldsymbol{B}^{(t)} := \left(\boldsymbol{W}^{(t-1)}\right)^{\top} \left(\boldsymbol{M}^{(t)}\right)^{-1}, \\ &\boldsymbol{S}_{xz}^{(t)} := \boldsymbol{S}^{(t)} \left(\boldsymbol{B}^{(t)}\right)^{\top}, \\ &\boldsymbol{S}_{zz}^{(t)} := \boldsymbol{I} - \boldsymbol{B}^{(t)} \boldsymbol{W}^{(t-1)} + \boldsymbol{B}^{(t)} \boldsymbol{S}_{xz}^{(t)}. \end{aligned} \tag{19}$$

Proposition 4.1: The inequality (17) always holds at every iteration in Algorithm 2.

(The proof of Proposition 4.1 can be seen in the longer version of our work [17].) This proposition guarantees the monotonic decrease of the objective value $J_{\mathrm{FA}}(\mathcal{H}^{(t)}, \boldsymbol{W}^{(t)}, \boldsymbol{\psi}^{(t)})$ during optimization.

5. Statistical Interpretation

As described in [10], FC-MKMC falls in a statistical framework. Concretely, FC-MKMC is an algorithm that performs the maximum likelihood estimation of a model parameter $\boldsymbol{M}$ of a probabilistic model $p_{\mathrm{FC}}(\boldsymbol{x} \mid \boldsymbol{M}) := \mathcal{N}(\boldsymbol{x}\,;\, \boldsymbol{0}, \boldsymbol{M})$, where $\boldsymbol{x}$ is an ℓ-dimensional random variate. In the statistical framework for FC-MKMC, maximum likelihood estimation is performed by finding the maximizer of the log-likelihood function

$$L_{\mathrm{FC}}(\boldsymbol{M}) := \sum_{k=1}^{K} \mathbb{E}_{q_k(\boldsymbol{v}_k)} \left[\log \mathcal{N}\left(\boldsymbol{v}_k\,;\, \boldsymbol{0}, \boldsymbol{M}_{v,v}^{(k)}\right)\right] \tag{20}$$

over the model parameter $\boldsymbol{M}_{v,v}^{(k)} \in \mathbb{S}_{++}^{n_k}$. Therein, $\boldsymbol{v}_k \in \mathbb{R}^{n_k}$ is the sub-vectorial variate in $\boldsymbol{x}_k$ associated with the visible objects in k-th data source; $\boldsymbol{M}_{v,v}^{(k)}$ is the submatrix of $\boldsymbol{M}$ associated with $\boldsymbol{v}_k$; and $q_k(\cdot)$ is the empirical distribution associated with the k-th data source such that the second moments satisfy

$$\mathbb{E}_{q_k(\boldsymbol{v}_k)} \left[\boldsymbol{v}_k \boldsymbol{v}_k^{\top}\right] = \boldsymbol{Q}_{v,v}^{(k)}. \tag{21}$$

From the log-likelihood function defined in (20), it is possible to derive an EM algorithm in which the E-step computes

the expected value $\mathbb{E}\left[\boldsymbol{x}_k\boldsymbol{x}_k^\top\right]$ through (8) and (9), based on the current model parameter. In the EM algorithm, the M-step updates the model parameter $\boldsymbol{M}$ through the maximizer of the expected complete-data log-likelihood function [15] over $\boldsymbol{M}$. However, the model matrix here is too flexible, and might overfit to the given empirical data.

5.1 EM Algorithm for PCA Model

Here, we present a connection between FC-MKMC algorithm and the classical statistical approach for missing value estimation. Let us discuss the case of replacing the full covariance model with the probabilistic principal component analysis (PPCA) model introduced in [16]. When employing PPCA model with the mean parameter fixed to zero, the probabilistic density of the n_k-dimensional random variate $\boldsymbol{v}_k$ is defined as

$$\begin{aligned} &p_{\mathrm{PCA}}\left(\boldsymbol{v}_k \,|\, \boldsymbol{W}, \sigma^2\right) \\ &:= \int \mathcal{N}\left(\boldsymbol{x}_k\,;\, \boldsymbol{0}, \boldsymbol{W}\boldsymbol{W}^\top + \sigma^2\boldsymbol{I}_\ell\right) d\boldsymbol{h}_k \\ &= \mathcal{N}\left(\boldsymbol{v}_k\,;\, \boldsymbol{0}, \boldsymbol{W}_v^{(k)}\left(\boldsymbol{W}_v^{(k)}\right)^\top + \sigma^2\boldsymbol{I}_{n_k}\right), \end{aligned} \tag{22}$$

where $\boldsymbol{W}_v^{(k)} \in \mathbb{R}^{n_k\times q}$ is the submatrix of $\boldsymbol{W}$ containing the rows associated with the visible objects. The log-likelihood function of this model is given by

$$L_{\mathrm{PCA}}\left(\boldsymbol{W}, \sigma^2\right) := \sum_{k=1}^{K} \mathbb{E}_{q_k(\boldsymbol{v}_k)}\left[\log p_{\mathrm{PCA}}\left(\boldsymbol{v}_k \,|\, \boldsymbol{W}, \sigma^2\right)\right], \tag{23}$$

which is used in finding the maximum likelihood estimate (MLE) of the model parameters $\boldsymbol{W}$ and σ^2 of the PPCA model. The expected complete-data log-likelihood function, also known as the Q-function, can be written as

$$\begin{aligned} &Q_t^{\mathrm{PCA}}(\boldsymbol{W}, \sigma^2) := \sum_{k=1}^{K} \mathbb{E}\left[\log p_{\mathrm{PCA}}(\boldsymbol{x}_k \,|\, \boldsymbol{W}, \sigma^2)\right] \\ &= -\frac{K}{2}\operatorname{logdet}\left(\boldsymbol{W}\boldsymbol{W}^\top + \sigma^2\boldsymbol{I}\right) \\ &\quad - \frac{1}{2}\left\langle \left(\boldsymbol{W}\boldsymbol{W}^\top + \sigma^2\boldsymbol{I}\right)^{-1}, \sum_{k=1}^{K}\mathbb{E}\left[\boldsymbol{x}_k\boldsymbol{x}_k^\top\right]\right\rangle, \end{aligned} \tag{24}$$

where we have dropped the terms that do not depend on the model parameters. Therein, the operator $\mathbb{E}$ takes the mathematical expectation under the joint posterior densities $q(\boldsymbol{h}_k|\boldsymbol{v}_k)q(\boldsymbol{v}_k)$ defined from the current value of $\boldsymbol{W}$ and σ^2. By letting $\boldsymbol{Q}^{(k)} := \mathbb{E}\left[\boldsymbol{x}_k\boldsymbol{x}_k^\top\right]$, the negative Q-function is equal to J_{PCA} up to constants, implying that the M-step of the EM algorithm is given by (12) and (13). Hence, we can say that the PCA-MKMC algorithm presented in the previous section is an EM algorithm.

5.2 EM Algorithm for FA Model

This section is concluded by showing that FA-MKMC is an EM algorithm for fitting the probabilistic factor analysis (PFA) model [18]. In the PFA model, a latent variable vector $\boldsymbol{z}_k \in \mathbb{R}^q$, drawn from the spherical Gaussian $\mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_q)$, is introduced for each data source. Then, $\boldsymbol{x}_k$ is generated by the process $\boldsymbol{x}_k = \boldsymbol{W}\boldsymbol{z}_k + \boldsymbol{\epsilon}_k$, where $\boldsymbol{\epsilon}_k$ is a Gaussian noise drawn from $\mathcal{N}(\boldsymbol{0}, \operatorname{diag}(\boldsymbol{\psi}))$. For this FA model, we treat $(\boldsymbol{x}_k, \boldsymbol{z}_k)$ as the complete data for k-th data source to develop an EM algorithm for maximum likelihood estimation. The probabilistic density of the n_k-dimensional random variate $\boldsymbol{v}_k$ is obtained by marginalizing $\boldsymbol{h}_k$ and $\boldsymbol{z}_k$ out from the joint density of the complete data:

$$\begin{aligned} &p_{\mathrm{FA}}(\boldsymbol{x}_k, \boldsymbol{z}_k \,|\, \boldsymbol{W}, \boldsymbol{\psi}) \\ &\qquad := \mathcal{N}\left(\boldsymbol{x}_k\,;\, \boldsymbol{W}\boldsymbol{z}_k, \operatorname{diag}(\boldsymbol{\psi})\right)\mathcal{N}\left(\boldsymbol{z}_k\,;\, \boldsymbol{0}, \boldsymbol{I}_q\right). \end{aligned} \tag{25}$$

The Q-function is written as

$$\begin{aligned} Q_t^{\mathrm{FA}}(\boldsymbol{W}, \boldsymbol{\psi}) &:= \sum_{k=1}^{K}\mathbb{E}\left[\log p_{\mathrm{FA}}(\boldsymbol{x}_k, \boldsymbol{z}_k \,|\, \boldsymbol{W}, \boldsymbol{\psi})\right] \\ &= -\left\langle \sum_{k=1}^{K}\mathbb{E}\left[\boldsymbol{z}_k\boldsymbol{x}_k^\top\right], \operatorname{diag}(\boldsymbol{\psi})^{-1}\boldsymbol{W}\right\rangle \\ &\quad - \frac{1}{2}\left\langle \sum_{k=1}^{K}\mathbb{E}\left[\boldsymbol{z}_k\boldsymbol{z}_k^\top\right], \boldsymbol{W}^\top\operatorname{diag}(\boldsymbol{\psi})^{-1}\boldsymbol{W}\right\rangle \\ &\quad - \frac{1}{2}\left\langle \sum_{k=1}^{K}\mathbb{E}\left[\boldsymbol{x}_k\boldsymbol{x}_k^\top\right], \operatorname{diag}(\boldsymbol{\psi})^{-1}\right\rangle \\ &\quad - \frac{K\ell}{2}\log(2\pi) - \frac{K}{2}\sum_{i=1}^{\ell}\log\psi_i, \end{aligned} \tag{26}$$

where $\mathbb{E}$ here is the mathematical expectation that operates under the joint posterior densities $q_t(\boldsymbol{z}_k, \boldsymbol{h}_k|\boldsymbol{v}_k)q(\boldsymbol{v}_k)$ depending on the current value of $\boldsymbol{W}$ and $\boldsymbol{\psi}$ obtained at the $(t-1)$ iteration. It can be shown that, by letting $\boldsymbol{Q}^{(k)} = \mathbb{E}\left[\boldsymbol{x}_k\boldsymbol{x}_k^\top\right]$, the expected values computed in the t-th iteration in the EM algorithm are expressed as

$$\sum_{k=1}^{K}\mathbb{E}\left[\boldsymbol{z}_k\boldsymbol{z}_k^\top\right] = K\boldsymbol{S}_{zz}^{(t)} \quad \text{and} \quad \sum_{k=1}^{K}\mathbb{E}\left[\boldsymbol{z}_k\boldsymbol{x}_k^\top\right] = K(\boldsymbol{S}_{xz}^{(t)})^\top. \tag{27}$$

In the M-step of EM algorithm, the model parameters $(\boldsymbol{W}, \boldsymbol{\psi})$ that maximize the Q-function are found. By setting the derivative of the Q-function with respect to the model parameters to zero, it turns out that the optimal factor loading matrix $\boldsymbol{W}$ and the noise variance vector $\boldsymbol{\psi}$ are given by (18). Hence, FA-MKMC is an EM algorithm. (The derivations of E-step and M-step can be seen in the longer version of our work [17].)

6. Experimental Settings

To test how much information the kernel matrices will retain after the completion processes, we subject the completed kernel matrices to a classification task: the functional

Table 1 Classification performance of the completion methods for 20% missed kernel data. The table entries are the ROC scores for a given functional class, averaged over ten trials. Here, the SVM classifier is trained on 20% of the combined completed kernel matrices. The boldfaced values correspond to the largest ROC score in each row, while the underlined values correspond to the ROC scores with no significant difference from the highest ROC score in each class.

Class	zero-SVM	mean-SVM	FC-MKMC	PCA-GK	PCA-K	FA-GK	FA-K
1	0.7914	0.7915	0.7995	0.8015	**0.8022**	0.8010	0.8006
2	0.7918	0.7925	0.7975	0.8025	**0.8032**	0.8014	0.8021
3	0.7941	0.7933	0.8000	0.8045	**0.8052**	0.8029	0.8032
4	0.8418	0.8431	0.8497	0.8529	**0.8534**	0.8516	0.8519
5	0.8839	0.8844	0.8956	0.8972	**0.8979**	0.8961	0.8967
6	0.7665	0.7669	0.7745	0.7780	**0.7783**	0.7770	0.7770
7	0.8321	0.8328	0.8414	0.8437	**0.8444**	0.8429	0.8440
8	0.7336	0.7336	0.7354	0.7407	**0.7418**	0.7391	0.7386
9	0.7621	0.7630	0.7651	0.7706	**0.7714**	0.7694	0.7695
10	0.7441	0.7445	0.7485	0.7551	**0.7570**	0.7525	0.7556
11	0.5766	0.5757	**0.5825**	0.5791	0.5807	0.5793	0.5772
12	0.9357	0.9347	0.9435	0.9448	**0.9453**	0.9443	0.9444
13	0.6818	0.6845	0.6794	**0.6913**	0.6911	0.6840	0.6838

classification prediction of yeast proteins. For this task, a collection of six kernel matrices representing different data types is used: the enriched kernel matrix $\boldsymbol{K}_{\text{Pfam}}$; the three interaction kernel matrices $\boldsymbol{K}_{\text{Gen}}$, $\boldsymbol{K}_{\text{Phys}}$, and $\boldsymbol{K}_{\text{TAP}}$; a Gaussian kernel defined directly on gene expression profiles $\boldsymbol{K}_{\text{Exp}}$; and the Smith-Waterman matrix $\boldsymbol{K}_{\text{SW}}$; as described in [6]. While each kernel representation contains partial information on the similarities among yeast proteins, the combination of these kernel matrices is known to provide a bigger picture of the relationships among these proteins through the different views of the data [1], [6], [7]—making the combined form more suitable in the overall predictions for the functional classification task.

Meanwhile, the 13 functional classes considered are listed in [6], which include metabolism, transcription, and protein synthesis, among others. If, for example, a certain protein is known to carry out metabolism and protein synthesis, then this protein is labeled as $+1$ in these categories and -1 elsewhere. This setting can then be viewed as 13 binary classification tasks.

In this study we utilized $K = 6$ related data sources for functional classification prediction of $\ell = 3,588$ yeast proteins. Initially, the kernel matrices may have missing rows and columns, which correspond to some missing information about the relationships among yeast proteins in the data sources. Our goal is to infer the missing entries in the kernel matrices, whilst retaining as much valuable information about the protein relationships as possible.

Our experiments consist of two stages: the kernel matrix completion (or missing data inference) stage, and the classification stage—the details of which are given in the subsequent sections.

6.1 Data Inference Stage

In this stage, mutual completion of the kernel matrices is performed. Since our data set has no missing entries, we generated incomplete kernel matrices by artificially removing some entries, following the process in [10]. Here, rows and (corresponding) columns were randomly picked, and undetermined values (zeros for zero-imputation method, and unconditional mean for mean-imputation method) were imputed; the details of which are referred to in [10]. For numerical stability of the two EM-based methods, $\boldsymbol{S}$ is transformed to $(K\boldsymbol{S} + 10^{-3}\boldsymbol{I})/(K + 10^{-3})$ at each iteration, a trick that is often used in Gaussian fitting. In our experiments, different percentages of missing entries were considered, and the incomplete kernel matrices were initialized by zero-imputation before proceeding with the completion processes, as specified in Alg. 1.

6.2 Classification Stage

After the completion process, a support vector machine (SVM) [2], [19] is used to predict whether a yeast protein belongs to a certain functional class or not. Since a yeast protein is not limited to a single functional class, the prediction problem is structured as 13 binary classification tasks, where an SVM classifier is trained on 20% randomly-picked data points on the combined kernel matrices. We then assess, in each functional class, the classification performance of the algorithms via receiver operator characteristic (ROC)—a widely-used performance measure for imbalanced data sets. Higher ROC score means better classification performance. The experiments were performed ten times, and the averages of ROC scores across the ten trials were recorded.

7. Experimental Results

In this section we present experimental comparisons among the five multiple kernel completion techniques: zero-SVM, mean-SVM, FC-MKMC, PCA-MKMC, and FA-MKMC. We refer to the completion methods zero-imputation and mean-imputation as zero-SVM and mean-SVM, respectively, after an SVM classifier has been trained. In our experiments, we used two criteria in choosing the number q of principal components for PCA and FA models:

the Guttman-Kaiser and the Kaiser criterion, where q is the number of all eigenvalues greater than the mean of the eigenvalues, and greater than one, respectively [20]. Henceforth, we will use PCA-GK and PCA-K to refer to PCA-MKMC, and FA-GK and FA-K to refer to FA-MKMC, with principal components via Guttman-Kaiser criterion and Kaiser criterion, respectively.

In the case of completion of 20% missed data, the ROC scores of the completion methods are summarized in Table 1. Here, the proposed method of restricting the model covariance achieves the highest ROC score in all classes, except at the 11th functional class where FC-MKMC obtains the highest ROC score; however, in this case, PCA-GK and PCA-K has no statistical difference from FC-MKMC according to a one-sample t-test. It can also be noted that in most cases, the classification performances of the restricted covariance models are not significantly lower than the highest ROC scores.

8. Concluding Remarks

In this study we present new methods, called PCA-MKMC and FA-MKMC, to solve the problem of mutually inferring the missing entries of kernel matrices, while controlling the flexibility of the model. In contrast to the full-covariance model parameter in the existing method, our algorithm imposes a restriction to the model covariance, capturing only the most relevant information in the data set through the principal components or factors of the combined kernel matrices. Our proposed method of restricting the model covariance matrix via probabilistic PCA and factor analysis resulted to significant improvements in the generalization performance, as shown in our empirical results for the functional classification prediction task in yeast proteins.

Besides kernel matrix completion, general matrix completion algorithms were developed by several studies [21]–[23]. The major difference of these studies from our work is that our algorithm infers the missing entries from multiple incomplete kernel matrices, whereas the existing methods complete a single incomplete matrix without any other auxiliary data. Many of these studies attempt to minimize the rank of the completed matrix, based on the justification of the low-rank assumption that has been discussed in many literature (e.g. [24]). Our new models, PCA-MKMC and FA-MKMC, are also derived from the low-rank assumption; this observation suggests that the reason why our new models performed better than the full-covariance model is due to the low-rank restriction incorporated in the new models.

It is also noteworthy that the goal of this study is the development of completion algorithms for general purposes that are not limited to classification, although it is possible to consider specializing the kernel completion algorithm to a particular purpose (such as classification) by adding some new loss functions. For such modified models, other optimization algorithms must be developed newly since the developed optimization algorithms work only for the current formulation.

Multiple model matrices may also be considered when there are many incomplete kernel matrices that differ highly from each other. In such scenario, we may employ a mixture-of-Gaussian model to generate multiple model matrices, allowing us to develop another EM algorithm for model fitting and matrix completion. We leave the development of such variants in future work.

Acknowledgments

This work was supported by JSPS KAKENHI Grant Number 40401236.

References

[1] G.R.G. Lanckriet, T. De Bie, N. Cristianini, M.I. Jordan, and W.S. Noble, "A statistical framework for genomic data fusion," Bioinformatics, vol.20, no.16, pp.2626–2635, Nov. 2004. http://dx.doi.org/10.1093/bioinformatics/bth294.

[2] B. Schölkopf and A. Smola, Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond, Adaptive Computation and Machine Learning, MIT Press, Cambridge, MA, USA, Dec. 2002.

[3] J. Shawe-Taylor and N. Cristianini, Kernel Methods for Pattern Analysis, Cambridge University Press, Cambridge, UK, 2004.

[4] M. Gönen, "Bayesian efficient multiple kernel learning," 29th International Conference on Machine Learning, pp.91–98, 2012.

[5] M. Deng, T. Chen, and F. Sun, "An integrated probabilistic model for functional prediction of proteins," Journal of Computational Biology, vol.11, no.2-3, pp.463–475, 2004.

[6] G. Lanckriet, M. Deng, N. Christianini, M.I. Jordan, and W.S. Noble, "Kernel-based data fusion and its application to protein function prediction in yeast," Pacific Symposium on Biocomputing 9:300–311, 2004.

[7] W.S. Noble and A. Ben-Hur, Integrating Information for Protein Function Prediction, ch. 35, pp.1297–1314, Wiley-VCH Verlag GmbH, Weinheim, Germany, Feb. 2008.

[8] T. Kin, T. Kato, and K. Tsuda, "Protein classification via kernel matrix completion," in Kernel Methods in Computational Biology, ch. 3, pp.261–274, The MIT Press, 2004. In B. Schölkopf, K. Tsuda, and J.P. Vert (eds).

[9] T. Kato, K. Tsuda, and K. Asai, "Selective integration of multiple biological data for supervised network inference," Bioinformatics, vol.21, no.10, pp.2488–2495, 2005.

[10] R. Rivero, R. Lemence, and T. Kato, "Mutual kernel matrix completion," IEICE Transactions on Information & Systems, vol.E100-D, no.8, pp.1844–1851, Aug. 2017.

[11] S. Bhadra, S. Kaski, and J. Rousu, "Multi-view kernel completion," Machine Learning, vol.106, no.5, pp.713–739, May 2017.

[12] A.P. Dempster, N.M. Laird, and D.B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," Journal of the Royal Statistical Society, Series B (Methodological), vol.39, no.1, pp.1–38, 1977.

[13] T. Matsuzawa, R. Relator, J. Sese, and T. Kato, "Stochastic dykstra algorithms for metric learning with positive definite covariance descriptors," The 14th European Conference on Computer Vision (ECCV2016), vol.9910, pp.786–799, 2016.

[14] J.V. Davis, B. Kulis, P. Jain, S. Sra, and I.S. Dhillon, "Information-theoretic metric learning," Proceedings on International Conference on Machine Learning, pp.209–216, ACM, 2007.

[15] G.J. McLachlan and T. Krishnan, The EM algorithm and extensions, 2nd Edition, Wiley series in probability and statistics, Wiley, Hoboken, NJ, 2008.

[16] M.E. Tipping and C.M. Bishop, "Mixtures of probabilistic principal component analyzers," Neural Comput, vol.11, no.2, pp.443–482,

Feb. 1999.

[17] R. Rivero and T. Kato, "Parametric models for mutual kernel matrix completion," April 2018. arXiv:1804.06095v1.

[18] D. Bartholomew, F. Steele, J. Galbraith, and I. Moustaki, Analysis of Multivariate Social Science Data, Second Edition, Chapman & Hall/CRC Statistics in the Social and Behavioral Sciences, Taylor & Francis, 2008.

[19] N. Cristianini and J. Shawe-Taylor, An introduction to support vector machines and other kernel-based learning methods, Cambridge university press, 2000.

[20] I. Jolliffe, Principal Component Analysis, Springer Verlag, 1986.

[21] J.-F. Cai, E.J. Candès, and Z. Shen, "A singular value thresholding algorithm for matrix completion," SIAM J. Optimization, vol.20, no.4, pp.1956–1982, March 2010.

[22] E.J. Candés and B. Recht, "Exact matrix completion via convex optimization," Found. Comput. Math., vol.9, no.6, pp.717–772, Dec. 2009.

[23] R.H. Keshavan, S. Oh, and A. Montanari, "Matrix completion from a few entries," CoRR, vol.abs/0901.3150, 2009.

[24] M.W. Berry, Z. Drmac, and E.R. Jessup, "Matrices, vector spaces, and information retrieval," SIAM Rev., vol.41, no.2, pp.335–362, June 1999.

Rachelle Rivero received her B.S. Mathematics and M.S. Applied Mathematics degrees from the University of the Philippines, Diliman (UP Diliman) in Quezon City, Philippines in 2007 and 2012, respectively. She is with the Institute of Mathematics, College of Science in UP Diliman as a lecturer from 2007 to 2008, and as an instructor from 2008 up to present. She is currently on study leave in UP to pursue her PhD degree in Japan, under the Japanese Government Scholarship (MEXT) program. She entered the Graduate School of Science and Technology in Gunma University in 2014 as a research student, and started her PhD degree in 2015 under the supervision of Prof. Kato. Her current interests include data fusion and bioinformatics.

Tsuyoshi Kato Tsuyoshi Kato received his B.E., M.E., and PhD degrees from Tohoku University, Sendai, Japan, in 1998, 2000, and 2003, respectively. From 2003 to 2005, he was with the National Institute of Advanced Industrial Science Technology (AIST) as a postdoctoral fellow in the Computational Biology Research Center (CBRC) in Tokyo. From 2005 to 2008, he was an assistant professor at the Graduate School of Frontier Sciences, University of Tokyo. From 2008 to 2010, he was an associate professor at the Center for Informational Biology, Ochanomizu University. He is now an associate professor at the Graduate School of Science and Technology, Gunma University. His current scientific interests include pattern recognition, computer vision, water engineering and bioinformatics. He is a member of IEICEJ.